



Example based style classification

Welnicka, Katarzyna; Bærentzen, Jakob Andreas; Aanæs, Henrik; Larsen, Rasmus

Published in:

Proceedings of the MICCAI workshop on Mesh Processing in Medical Image Analysis (MeshMed)

Publication date:

2011

[Link back to DTU Orbit](#)

Citation (APA):

Welnicka, K., Bærentzen, J. A., Aanæs, H., & Larsen, R. (2011). Example based style classification. In *Proceedings of the MICCAI workshop on Mesh Processing in Medical Image Analysis (MeshMed)*
<http://www2.imm.dtu.dk/projects/MeshMed/>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Example based style classification

Katarzyna Welnicka, Jakob Andreas Bærentzen, Henrik Aanæs, and Rasmus Larsen¹

DTU Informatics, Technical University of Denmark

Abstract. We address the problem of analysis of families of shapes which can be classified according to two categories: the main one corresponding usually to the coarse shape which we call the function and the more subtle one which we call the style. The style and the function both contribute to the overall shape which makes the general analysis and retrieval of such shapes more challenging. Also there is no single way of defining the style as this depends much on the context of the family of shapes used for the analysis. That is why the definition needs to be given through the examples.

The straight forward way of finding the shape descriptors 'responsible' for a given category would be to use well known statistical methods and find through them such descriptors with which we are able to classify shapes according to a given category. When a function is dominating this approach might not suffice - we might be unable to find a set of descriptors which are independent of a given function. We show how to decouple the effect of the style from that of the function by considering the shapes of the same function but different styles. We also propose a metric coanalysis approach: if two styles are similar this similarity should be reflected across different functions.

We show the usability of our methods first on the example of a number of chess sets which our method helps sort. Next, we investigate the problem of finding a replacement for a missing tooth given a database of teeth.

1 Introduction

While digital shapes are starting to have a number of medical applications, for instance related to hearing aid production and dental work, the use of digital shapes does not necessarily lead to complete automation. Typically, certain procedures are still left to human operators. However, it is an important goal to be able to help the human operator as much as possible. The particular scenario which we address in this paper is the selection of tooth shapes which can serve as the starting point for digital models of crowns.

There is a lot of work in the shape analysis and especially shape retrieval community with a task of finding the most similar shape to a query one. However, many shapes might be classified not only according to a single category, e.g. as being a table or a chair, which we will call the function, but also according to the style: A table and a chair of the same style share common geometric properties which are different from the overall shape. The style and the function both

interact and contribute to the overall shape of the object. It is not always easy to separate them and point out geometric elements responsible for a function or a style.

The general distinction between the specific shape properties, which tells which ones are responsible for the style and which for the function, is not possible as this depends on a context. That is why we define the style and function through examples.

1.1 Existing Work related to style function recognition

Style and function separation in the context of man made three dimensional shapes was recently mentioned by Xu et al. [14], where the style of an object is defined by the proportions (anisotropic scaling) of its parts. It seems to be very intuitive and reasonable approach but this does not exhaust the subject. The style might be hidden in details, repetitions of some patterns or some other types of deformation as well. Very often it is hard to define it mathematically although the human brains usually do not have problems in recognizing it.

In many shape processing articles, even if the problem of style is not addressed in an explicit way there are situations where the space of given shapes is broken into two different independent classification systems. In the deformation transfer [10] different kinds of animals can take similar poses in which case it is quite easy to localize them, as the type of animal is described by an intrinsic metric of the shape surface, and the pose is its embedding in three dimensional space. The idea of geometric texture [1] fits within this framework as it aims to separate overall shape from its geometric details. Application of example based priors for surface reconstruction [9, 3] can also be seen as imposing style of the object.

In the image processing field Hertzmann et al. [6] presented a method that given three images, an image with style A and function 1, an image with style B and function 1, an image with style A and function 2, created an image with style B and function 2. The same concept was also explored by this group in the field of curve styles [7]. Other related problems can be present when dealing with images of fonts, separating lighting conditions from the scene and distinguishing between the spoken language and the accent - all of those three cases were examined through bilinear models by Tenenbaum et al. [12].

Tenenbaum's framework requires establishing one to one correspondences of the parts both for the style and with the function - for example fonts are compared through pixels of a bitmap: in general for different types of shapes obtaining such correspondences is usually hard to achieve. Similar correspondences need to be established across the styles for Hertzman's work. Our approach does not require any correspondence finding, which usually is a costly task and sometimes it is not possible as for example in the problem of registering a table to a chair. Instead we do shape comparisons through the shape descriptors. There is a lot of current work on content based shape retrieval and different descriptors might capture different properties of the shape and produce different notion of their similarity. So a good approach is to extract many different shape descriptors and combine them in a proper way.

1.2 Metric learning

If the feature space is available, many well established statistical methods can be used such as Linear Discriminant Analysis [8] which modifies the feature space so that, for a given training set containing objects from different classes, it maximizes intra class variance and minimizes within class variance. Similar approach was also used by [13] which gives the possibility of defining the similarity and dissimilarity relationships between selected pairs of objects.

As mentioned by Giorgi et al. [5] for the case of shapes there are many useful shape descriptors like skeletons, trees, weighted point sets, which do not provide multidimensional feature space. Still with such descriptors there is usually a way of establishing a notion of similarities between different shapes which results in some kind of pseudodistance.

Giorgi et al. [5] customize a way of combining a set of distances between shapes so that user defined similarity is captured. In this work the metric is modified in order to reflect the user defined constraints of nearby or far away shapes. The final metric is taken as a maximum distance from distances given by all of the metrics, however the particular metrics are scaled according to a similarity feedback provided by the user.

The approach of combining different metrics relies on the fact that at least there exists a set of shape descriptors which can capture the similarity imposed by virtue of shared stylistic or functional properties. For function, which usually is easier to distinguish such an approach would be very suitable. However when a style needs to be extracted it might not be enough and not even single descriptor might exist which is purely responsible just for the style.

One of our main observations concerning this problem is that knowing what is the function of an object enhances the possibilities for style recognition. For many descriptors information on style is coupled with information on function. In general, when the distance between two shapes is small, it might be both due to similarity in the style and similarity in the function. The retrieval of style related information can be achieved when providing a set of shapes sharing the function and having different styles.

The requirement of recognizing the object of the same function or the same style as being close is not enough in such case. We also want our dissimilarity measures between shapes to be consistent across different functions. This requirement stems from the fact that we want to be able to find the most similar styles and most similar functions. However, for our style-function task case we do not have a direct input which indicates which styles are similar and which are not. Instead we have some notions of similarities which are induced by different shape descriptors and there is a need to choose the ones which are relevant. This relevance is not defined directly by indicating the shapes which should be treated as similar but indirectly as a consistency requirement: dissimilarity or similarity between the styles should be reflected in a similar way for different functions.

Similar indirect consistency approach methodology can be found in [15] which removes incorrect mappings of sets of different views. The assessment of the quality view mappings is done through analyzing them in broader context of

the consistent mapping loops. If the loop is inconsistent it means that one of the mappings that belongs to it is wrong and the consistent loop means that mappings are likely to be correct. Having evaluated the correctness of many loops the bad mappings are spotted through a loopy belief propagation.

1.3 Contribution

This paper focuses on an issue, which we think has many application areas, but was not very much explored yet: the analysis and classification of shapes according to more than one category, when categories may be coupled together which in our case is the style and the function.

We propose here a general methodology which can be applied in order to deal with the style-function determination problem. Because the style and the function strongly depend on the context, defining it by providing example shapes seems to be the most general approach.

We show the method for decoupling the effect of the style from that of the function. By having as a training dataset the shapes of the same function and different styles, we can factor out the function and determine the most likely style of an unknown shape as the closest shape from the set. In an analogous way by using the shapes of the same style but different functions the unknown function may be retrieved.

We realize that the key to success is to find a good metric between the shapes: metric which can capture both stylistic and functional features. Using the example of chess pieces we show what are the desired properties of such a metric (section 2) and how to decouple the style from the function when only one metric is available.

We also show how to find an appropriate metric by combining the metrics obtained through different shape descriptors (section 3). Novel in our case is that we do not only use standard similarity notions but also explore the metric consistency approach. The problem is illustrated with the example for a tooth dataset.

After the example of chess pieces, we focus on teeth as an example medical application. Note that our framework is fairly generic. It could be applied to any type of biological surface which exhibits variation due to both style and function.

2 Decoupling metric

In this section we will show how an information about style and function hidden in the same metric can be decoupled. This is illustrated by the example of style - function classification based on the chess pieces. Since the chess pieces are rotationally symmetric, their three dimensional representation can be reduced to the space of plane curves by taking the outline curve obtained through rotating the chess piece by the rotational symmetry axis. The Translation Invariant Dynamic Time Warping [2] is used in order to establish a similarity metric $d(.,.)$ between the objects.

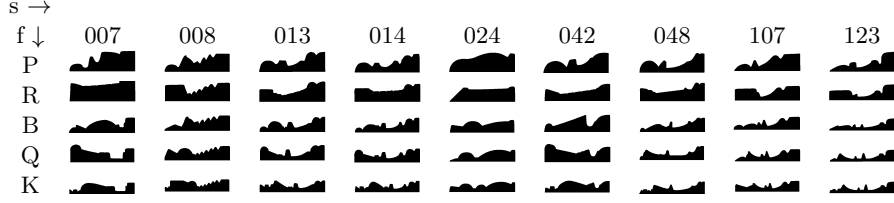


Table 1. The outline curves of the chess pieces. Our dataset has 45 chess pieces, which are the scans taken from 9 existing chess sets. The function is the type of the chess piece (pawn, rook, bishop, king, queen) and the style is the set the chess piece belongs to.

Fig. 1. Similarities between the curves. Each block has the same function or style, the diagonals of blocks are darker which reflects the smaller distance when the function or the style is the same.

2.1 Likelihoods computation

In our setup the proximity of two shapes can be affected by two factors: the similarity of the style and the similarity of the function. Also dissimilarity with respect to one factor, which usually is a style might be more subtle than the other one. However if we have in the training set pieces which share the same style (or function) but have different function (style), then it is possible that we may factor the style (function) out. Instead of taking absolute distances one may use the relative distance information: the difference of the distances. For example if the distance to a king is smaller than a distance to a bishop of the same style we may say that the unknown shape is more likely to be the king than to be something else and that will affect the sign of the distance difference.

The partial likelihood of unknown shape x to be a function K , when we have two example shapes of the same style S_i of which one (denoted as KS_i) is of a function K and other one NS_i is of a function other than K , is equal to:

$$l_f^{S_i, N}(x, K) = d(NS_i, x) - d(KS_i, x).$$

In our training dataset TS_i , for a given style S_i , we may have more than just one shape not being of a function K so we take the mean plus the minimum of all of the partial likelihoods:

$$l_f^{S_i}(x, K) = \text{mean}_{K \neq NS_i \in TS_i} l_f^{S_i, N}(x, K) + \min_{K \neq NS_i \in TS_i} l_f^{S_i, N}(x, K).$$

Note that minimum is equal to the distance to the closest of the known shapes from style S_i other than KS_i , minus the distance to KS_i . If the function K is the closest of the shapes from that style, then the minimum will be positive otherwise it will be negative. The mean value stabilizes the results by taking into account distance measures of all of the shapes of this style.

In order to gather the information from all of the training styles we take the mean value plus the maximum of all the styles, for which in a training set there is a function K and some shapes not being of function K .

$$l_f(x, K) = \max_{\substack{S_i \in S, KS_i \in TS_i, \\ K \neq NS_i \in TS_i}} l_f^{S_i}(x, K) + \text{mean}_{\substack{S_i \in S, KS_i \in TS_i, \\ K \neq NS_i \in TS_i}} l_f^{S_i}(x, K).$$

Here by taking the maximum we are favoring the style for which the K function is most likely. The mean is again added to get the distance information from all known styles.

There might be cases when we do not have enough information in the training set for establishing likelihoods. This happens when there is no set which has a training representative for the function K and for some shape which is not of a function K . In such a case we set the likelihood to zero.

The likelihood computation of "x being the style i" is done in an analogous manner. Then for a given x the cost of assigning to it style j and function i is equal to: $l(x, F_i, S_j) = l_f(x, F_i) + l_s(x, S_j)$.

2.2 Chess classification example with the assignment problem

We use the likelihoods as negative costs and solve the minimum linear assignment problem for the unknown labels and loose chess pieces.

Table 2 contains the results of the assignment problem if the training dataset is one set and one function, and we are searching for other chess pieces. The results depend a lot on the type of the set and function imposed as an example shape. Some of the sets contain a lot style and function information but some other do not. The sets 024 and 008 are performing the worst also the results for the rooks is always worse than for other functions.

2.3 Multiple step assignment

An assignment problem with the costs defined above does not make use of the information about all of the distances between the shapes. If we are able to locate the chess pieces of which we can expect that the initial matching went correctly we can add those into a training dataset with the labels obtained by the

	P	R	B	Q	K	mean
123	9	24	12	17	18	18
f	13	11	5	9	10	9.6
s	13	21	9	10	10	12.6
107	17	20	16	12	17	16.4
f	13	12	10	5	9	9.8
s	12	15	11	7	10	11
014	20	20	10	9	14	14.6
f	6	11	2	2	6	5.4
s	17	13	8	7	10	11
024	28	27	19	25	28	25.4
f	20	22	15	20	22	19.8
s	16	12	8	12	14	12.4
013	14	24	6	6	19	13.8
f	7	11	2	2	8	6
s	8	15	4	4	13	8.8
048	16	17	11	10	17	14.2
f	7	6	4	2	8	5.4
s	14	12	8	8	12	10.8
008	22	24	19	16	24	21
f	18	21	15	10	17	16.2
s	14	18	9	7	11	11.8
042	20	24	21	14	18	19.4
f	14	14	13	9	10	12
s	11	16	9	7	12	11
007	15	23	8	10	21	15.4
f	8	12	6	7	12	9
s	14	15	4	6	16	11
mean	19	22.56	13.56	13.22	19.56	17.58
f	11.78	13.33	8	7.33	11.33	10.36
s	13.22	15.22	7.78	7.56	12	11.16

Table 2. Mismatches of the single assignment problem with one style and one function given. The table contains the general number of pieces with mismatched total label, mismatched function and the mismatched style.

initial assignment. In order to estimate the labeling reliability, we calculate the diagonal cost of the assignment which we define as the average sum of similarities between all the pieces having the same style or function labels. For a hypothetical unknown chess piece we might add it for a moment to the training set and calculate what is the diagonal cost when assignment is solved with the use of this piece. We discovered that instead of calculating diagonal costs directly it is better do the inverse assignment, which is performed by swapping the unknown data with the known and then calculating the diagonal cost. Then the smaller the inverse diagonal cost is the more reliable is the hypothetical assignment of the unknown chess piece to its label.

In order to minimize bad choices we always take the piece having minimum inverse diagonal cost and is reliable according to the additional reliability criteria. We add it to the initial dataset and repeat the assignment and the most reliable pieces addition until there is no reliable piece to be added. Then we use the assignment from the last step as the final assignment.

In the results (table 3) we observe an average improvement of the assignment tasks by approximately 3 chess pieces. Usually if initial guess is quite good but not perfect then correctness of the matching may be improved quite well. If there are too many mismatches the improvement does not occur: as then we also take as reliable the matchings which are not correct. Usually it does not make the solution worse but keeps it at a similar level as it was with the initial problem.

3 Finding the good metric

The case of chess pieces was special problem as we were able to reduce the shape information to the space of the curves and had a way of establishing

	P	R	B	Q	K	mean
123	19	23	9	7	15	14.6
f	11	11	4	2	9	7.4
s	10	18	5	5	10	9.6
107	15	14	10	5	11	11
f	6	7	6	0	5	4.8
s	13	13	5	5	8	8.8
014	14	12	5	4	17	10.4
f	8	8	0	0	9	5
s	9	9	5	4	11	7.6
024	24	25	18	26	27	24
f	20	18	14	20	20	18.4
s	14	18	7	12	11	12.4
013	15	21	0	0	19	11
f	8	11	0	0	11	6
s	9	14	0	0	12	7
048	19	19	2	4	12	11.2
f	11	9	2	0	5	5.4
s	14	14	0	4	9	8.2
008	22	28	23	9	8	18
f	18	22	14	4	7	13
s	13	25	14	6	5	12.6
042	22	21	15	17	23	19.6
f	13	14	10	15	13	13
s	14	15	7	2	16	10.8
007	17	25	6	8	11	13.4
f	8	9	2	2	6	5.4
s	13	19	4	6	5	9.4
mean	18.56	20.89	9.78	8.89	15.89	14.8
f	11.44	12.11	5.78	4.78	9.44	8.71
s	12.11	16.11	5.22	4.89	9.67	9.6

Table 3. Mismatches of the multiple assignment problem with one style and one function given. The table contains the general number of pieces with the mismatched total label, mismatched function and mismatched style.

similarity between those curves by using Translation Invariant Dynamic Time Warping. In general for three dimensional shapes we do not know a good metric in advance, instead we have many propositions of metrics $d_i(,)$ which can be obtained through different kinds of shape descriptors D_i .

The task is to choose such a metric d_i or some combinations of metrics with which we can distinguish between different styles. As mentioned in the introduction it is not enough to be able for the objects of the same style to be close but also the dissimilarity measures should be consistent across different functions and we don't know which one will work best for specific problem.

This requirement can be illustrated with the problem of tooth shapes. Suppose a patient has one tooth destroyed. In order to be able to reproduce its shape, we want to find from a database a tooth which is mostly similar to the existing tooth he has. We have a molar missing but because a premolar is still in the patient's mouth, we wish to search in our database for a mouth which has the most similar premolar to the patient's. From that mouth we take a molar as a template for our new tooth. This approach assumes that similarity for premolars induces a similarity between molars.

This case shows that the metric consistency requirement is necessary as it aids in many concrete tasks - like searching for the closest to missing data. Here we do not know directly what 'close' means, as we have many metric but don't know which one is a correct. Usually a correct metric combination in such a case can be found by giving example pairs of shapes which are similar and which are dissimilar[5]. In our case we do not have such information. Instead we can impose the metric consistency requirement: the distances between shapes having different styles and function A should be close to the distances of the shapes of the same styles and function B .

3.1 Metric consistency

Assume we have a set of training shapes $F_{i=1..n_i} S_{j=1..n_j}$, where i indicates the function and j style. We also have a k_n potential distances $d_k(,)$

Let us take all distances $d_k(F_{i_1} S_{j_a=1..n_j}, F_{i_1} S_{j_b=1..n_j, j_b \neq j_a})$ between different shapes of the function i_1 . In order to be comparable those distances need to be normalized which we do by dividing them by the median from obtained distances. This results in a $\binom{n_j}{2}$ dimensional vector of k-distances between shapes with function i_1 which we will denote $v(d_k, f_{i_1})$.

For each pair $i_1 \neq i_2$ of two different functions we can establish the **consistency score** $cs_{d_k}^{i_1, i_2}$ with respect to a distance k and function i_1 and i_2 as a norm of difference of distance vectors:

$$cs_{d_k}^{i_1, i_2} = \sqrt{\sum_{l=1..(\frac{n_j}{2})} (v(d_k, f_{i_1})_l - v(d_k, f_{i_2})_l)^2}$$

In order to calculate total consistency factor (TCF_k) for a distance measure k sum of the differences for all function pairs is taken. Note that smaller TCF_k is the more consistent is d_k with respect to style.

We construct the final metrics by summing the metric obtained through different shape descriptors with weights that promote consistency.

$$D_f(,) = \sum_k e^{(-2 \frac{TCF_k}{mean(TCF)})} \frac{d_k(,)}{\sigma_{d_k}}$$

where σ_{d_k} median distance from distances $d_k(,)$ between all training shapes.

From the final metrics we can also compute consistency scores $cs_{D_f}^{i_1, i_2}$. This consistency measures can be used in order to asses what kind of tooth types are better correlated. For example two neighbor upper molars can be more correlated than molar and incisor. So if a molar is missing and we have the neighbor molar and incisor, we should give higher weight for query of closest mouth with respect to a molar than with respect to incisor. We can also compute mean distances between styles by summing $v(D_f, f_j)$ for all function types j .

3.2 The tooth problem



Fig. 2. Front view of molar, premolar and incisor from 3 different mouths

In the teeth analysis task we take a type of a mouth as style and a tooth type as function. An example dataset we use for this problems contains teeth shapes

(figure 2) from 6 different mouths. In order to make number of styles larger, we assume that the left side of a mouth will be treated separately from the right part. Thus we have 12 styles which we will label as A,B,C,D,E,F,a,b,c,d,e,f, where big letter means one left part of a mouth and small the other one. We have taken 10 tooth types 2 upper molars, lower molar, 2 upper premolars, lower premolar, upper canine, upper incisor, 2 down incisors. They are labeled and placed in the following order: 7M,6M,6m,5P,4P,4p,3C,1I,1i,2i, where upper case means respectively upper tooth.

In order to get independence of meshing we uniformly sampled the surface of teeth and computed descriptors out of those samples. We used local shape descriptors which rely on neighborhood at some distance from a given position. As neighborhood size we have taken 0.01 0.04 0.16 and 0.64 of the radius of a bounding sphere of a tooth. For slippage we used 0.01 0.04 and 0.16. In Total we had: 2x4 descriptors for main curvatures obtained by fitting primitives [11], 3x4 eigenvalues of covariance matrix of points sampled from the neighborhood area and 12x3 slippage coefficients [4] which are 6 eigenvalues of slippage covariance matrix and 6 is a translational contribution to its eigenvectors. We took 2 samples for 1000 points, for which soft histograms were computed. Histograms from two independent sampling were compared. The mean across all training shapes, of their difference was taken in order to estimate the measure error coming from different samplings. Then the mean of the 2 sample histogram is taken. However in order to compare two histograms for shapes S_i and S_j the distance between two bins is reduced by the previously computed measure error. Then the sum of those values is taken across all bins as our distance $d_k(S_i, S_j)$.

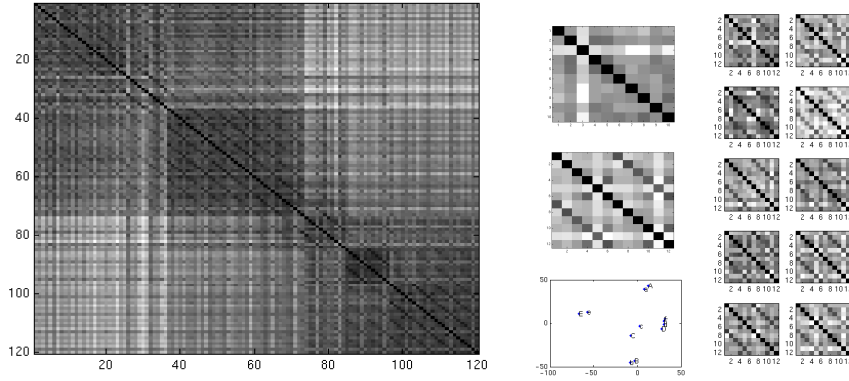


Fig. 3. Left: final metric D_f obtained with teeth database (indices grouped with respect to teeth type). Center: $cs^{i,j}$ for different teeth types, the average distance between the styles, and the multidimensional scaling plot for the average distances. Right: metrics between different styles when a function is fixed, obtained from final metric with the training styles $AFade$.

Then the total consistency factors are computed as mentioned in the previous section and the final metric is computed. Figure 3 contains the resulting metric, where all of the available teeth were used. It is worth mentioning that the consistency score for a resulting metric is smaller than the scores from any particular metrics. Note that the styles that come from the same mouth (left or right part) are being found as close. Also note that neighbor teeth tend to have more consistent scores. This information might be used when searching for a missing tooth. Let us consider the case when *AFade* styles were taken as training styles and a metric *T* was created. Then a patient comes with mouth of style *C* and with missing *4P*. We have scans of his *4p*, *5P* and *1i* and we have $cs_T^{4P,4p} = 1.6391$, $cs_T^{4P,5P} = 1.5819$ and $cs_T^{4P,1i} = 1.7543$, so we use tooth *5P* as it has the best consistency. We use $v(T, f_{5P})$ instead of unknown $v(T, f_{4P})$ in order to evaluate proximity between teeth (5th and 6th plot on the Right of figure 3). We evaluate distances between *C* and *AFade* among teeth of type *5P* and the mouths sorted with respect to distance will be *FdaAE* if we checked the ground truth we have *dFaEA*. Despite the swaps which was a result of a very close similarity values of *dF* and *EA* we can see that in general dissimilar teeth remain dissimilar.

We also tested on how the consistency properties of metric change when different subset of styles was used as training dataset. We generated metric from this information and evaluated the results on all of the data.

Usually removing only small number of mouths did not increase or even slightly decrease the consistency scores. Only when using 3 or 4 mouths, the results seemed be different. This might come from the fact that there was always some symmetric tooth left in the set which was able to set the consistency scores in a correct way. The increase was mostly noticeable when styles which are close to each other are used as training set (table 4).

training	TCS	training	TCS	training	TCS	training	TCS	training	TCS
<i>DdFf</i>	111.007	<i>EFb</i>	102.088	<i>cEF</i>	98.17	all	97.94	none	117.22
<i>Eef</i>	117.107	<i>ABd</i>	99.138	<i>ADEFc</i>	97.35	<i>AFade</i>	98.494	<i>ABCDFaf</i>	95.929

Table 4. Total consistency factors when using different mouth subsets as training data.

4 Conclusion

In this article we presented methods of working with shapes that can be classified into having two categories: style and function. One of them decouples style and function when they are incorporated into the same metric. The second finds a metric as a combination from existing ones when a consistency between different function types is needed. Those methods were illustrated by the chess and tooth datasets. We are aware that for a further analysis and development of our methods more data will be needed but we think the results obtained so far are promising.

Acknowledgments: We would like to thank 3Shape for the tooth dataset and for making their scanner available when scanning the chess pieces.

References

1. Vedrana Andersen, Mathieu Desbrun, J. Andreas Baerentzen, and Henrik Aanaes. Height and tilt geometric texture. In *Proceedings of the 5th International Symposium on Advances in Visual Computing: Part I*, ISVC '09, pages 656–667, Berlin, Heidelberg, 2009. Springer-Verlag.
2. Alon Efrat, Quanfu Fan, and Suresh Venkatasubramanian. Curve matching, time warping, and light fields: New algorithms for computing similarity between curves. *J. Math. Imaging Vis.*, 27(3):203–216, 2007.
3. Ran Gal, Ariel Shamir, Tal Hassner, Mark Pauly, and Daniel Cohen-Or. Surface reconstruction using local shape priors. In *SGP '07: Proceedings of the fifth Eurographics symposium on Geometry processing*, pages 253–262, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.
4. N. Gelfand and L. Guibas. Shape segmentation using local slippage analysis. In *Proc. Symp. Geom. Processing*, pages 219–228, 2004.
5. D. Giorgi, P. Frosini, M. Spagnuolo, and B. Falcidieno. 3d relevance feedback via multilevel relevance judgements. *Vis. Comput.*, 26:1321–1338, October 2010.
6. Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin. Image analogies. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340, New York, NY, USA, 2001. ACM.
7. Aaron Hertzmann, Nuria Oliver, Brian Curless, and Steven M. Seitz. Curve analogies. In *EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering*, pages 233–246, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.
8. G.J. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*. Wiley, 1992.
9. Mark Pauly, Niloy J. Mitra, Joachim Giesen, Markus Gross, and Leonidas J. Guibas. Example-based 3d scan completion. In *SGP '05: Proceedings of the third Eurographics symposium on Geometry processing*, page 23, Aire-la-Ville, Switzerland, Switzerland, 2005. Eurographics Association.
10. Robert W. Sumner and Jovan Popović. Deformation transfer for triangle meshes. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 399–405, New York, NY, USA, 2004. ACM.
11. Gabriel Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. pages 902–902. IEEE Computer Society, IEEE Computer Society, 1995.
12. Joshua B. Tenenbaum and William T. Freeman. Separating style and content with bilinear models. *Neural Comput.*, 12:1247–1283, June 2000.
13. Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15*, volume 15, pages 505–512, 2002.
14. Kai Xu, Honghua Li, Hao Zhang, Daniel Cohen-Or, Yueshan Xiong, and Zhiqian Cheng. Style-content separation by anisotropic part scales. *ACM Transactions on Graphics, (Proc. of SIGGRAPH Asia 2010)*, 29(5):to appear, 2010.
15. Christopher Zach, Manfred Klopschitz, and Manfred Pollefeys. Disambiguating visual relations using loop constraints. In *CVPR'10*, pages 1426–1433, 2010.